MINISTRY OF SOCIAL WELFARE

JERUSALEM, ISRAEL

RESEARCH & DEMONSTRATION PROJECT ON REHABILITATION OF BLIND, PARTIALLY SIGHTED AND OTHERWISE HANDICAPPED PERSONS AS DATA PROCESSING MACHINE OPERATORS AND COMPUTER PROGRAMMERS

REPORT NO. 7

By

EZRA SHAPIR

Prepared under a grant from the Social and Rehabilitation Administration, Department of Health, Education and Welfare,
Washington D.C., U.S.A.

VRA-ISR-13-64

Jerusalem, May 1969

- - - - - - - - - - -

c.2

MINISTRY OF SOCIAL WELFARE

in cooperation with

The Office Mechanisation Centre

The IBM World Trade Corporation (Israel Branch)

The Jewish Institute for the Blind

JERUSALEM, ISRAEL

RESEARCH & DEMONSTRATION PROJECT ON REHABILITATION OF BLIND, PARTIALLY SIGHTED AND OTHERWISE HANDICAPPED PERSONS AS DATA PROCESSING MACHINE OPERATORS AND COMPUTER PROGRAMMERS

Principal Investigator: Dov Chevion
Program Director: Yehuda Schiff

R E P O R T N O. 7

By

E Z R A S H A P I R

Technical Engineer - Research Consultant

Prepared under a grant from the Social and Rehabilitation Administration, Department of Health, Education and Welfare,
Washington D.C., U.S.A.

VRA-ISR-13-64

Jerusalem, May 1969

- - - - - - - - - - -

C O N T E N T S

FOREWORD

by Dov Chevion

Principal Investigator

Director, Office Mechanization Centre
Jerusalem, P.O. Box 13016

Report No. 7 summarizes another step forward in our research project in the rehabilitation of the blind, partially sighted and otherwise handicapped persons as computer operators and programmers.

Before we proceed to outline the future direction of the research, let us review the results hitherto achieved and their evaluation.

1. Selection tests have been employed and even special tests for the blind have been developed. The results do not yet allow us to draw any positive conclusions; in some cases they seem even to diverge. Our population for the test was too small, so that regularities and irregularities could be due to random deviations and cannot yet have sufficient significance.

2. The blind and partially sighted have been successfully taught computer programming. A detailed syllabus has been devised and a number of multi-sensory aids have been employed in order to enable the blind to grasp the contents of the course, especially in those cases where vision seemed to be important and even indispensable. All that is shown in detail in the report. A balanced combination of lessons in theory and practical work on the computer in the centre was no doubt among the factors which contributed to the success.

The main conclusion that may be drawn from our experience in that field is that it is possible to effectively teach blind, partially sighted and otherwise handicapped computer programming (provided that the selection method works satisfactorily, similarly as in the case of unhandicapped pupils). This statement does not seem limited to a special type of computer, although more and very advanced computer systems will provide new problems in teaching technique which will have to be overcome when teaching their programming to the blind.

The blind and the partially sighted are able to perform their work quite satisfactorily and independently of their unhandicapped supervisors or co-workers. However, the degree of independence is not yet as high as we would like to have it. The problems in that respect focus mainly in the stage of input (preparation of the programmes) and stage of output (checking and debugging). We have overcome all essential difficulties encountered by the blind; however, the solutions - as may be seen from reading the relevant parts of this report - are not yet final and their cost (e.g. computer time) in comparison with unhandicapped programmers is not yet low enough, although already it is relatively by far not expensive.

3. We have carefully studied the breakdown of the work of a computer operator and have found that it is possible to train handicapped who are paraplegic and amputees seated in wheelchairs as computer operators so that they reach a satisfactory level of performance in comparison with unhandicapped trainees. This can of course be achieved under certain conditions regarding some space requirements and some changes in the

method of transportation of boxes of continuous paper. Both conditions are relatively quite bearable in practice.

These are the three different aspects of the research which summarise its main content. When we consider the aim we have set for ourselves: to achieve through training, didactic technique, and teaching aids, a wholly independent blind programmer; we see that this has not yet been attained when compared with the results achieved hitherto. This points to the following main objectives to be studied in the forthcoming stages of the research:

(a) Reliability of the selection test will have to be ensured. This is especially important, and particularly difficult in respect to the personality and aptitude test of the blind and partially sighted persons. Additional experience with trainees of future courses will augment the population under research and might contribute towards more reliable and conclusive results.

(b) Improvement of the teaching methods - both of theory and of practical performance - with the aim of making the absorption of the instruction easier and more effective for the blind trainee. The development of multi-sensory aids, in addition to those described in this report, and the improvement of those already in use is necessary. One of the main advantages here may be the shortening of the period of instruction.

(c) Further development of special instruments in order to solve problems of input and debugging of programmes in a more effective way, and to enable the blind person to perform all his work independently.

(d) Development of general programmes for the translation of the usual output into Braille along with the design of a new Braille code specially adapted and suitable for computer processing will be the main ways in which to improve debugging technique and to lesses as far as possible the cost of the use of the computer for the translation into Braille.

These will be the main fields in which our future research will contrate, and we hope that the results will continue to be encouraging.

I should like to express our appreciation to the professional staff of the Centre, who by their perfect team work contributed to the preparation of this report. A great deal of technical work was involved here, and thanks in this respect are due to Mrs. Tamara Karlin and Mrs. Lora Huberman for their diligent work in typing and editing the report, and to Mr. David Rubinger who prepared the photographs.

SELECTION OF CANDIDATES

Time Magazine, August 18, 1967, in its article "The Software Snarl", says: "The company is running into trouble because of shortcomings in men, not in the machine. In just two decades the electronic marvels have grown so complex and intricate to operate that man is hard put to maintain the proper control. The problem is one of telling the machine what to do and how to do it."

All over the world, including Israel, more money is being invested for planning and programming than for buying or renting electronic computers. The situation in the computer market being such, we should examine its implications for those whose task it is to rehabilitate blind or otherwise handicapped in this field.

In the framework of this project we have proved that the blind are able to contribute in the field of computer programming. We now have to intensify our efforts to integrate them to the extent that they will be totally independent of their co-workers after they are placed in work. Only after we find a solution to this problem of dependency will we be able to state that the blind are ready to compete in the computer programming market on the basis of their knowledge and experience in the work and not on sentiment.

In this chapter we will deal with the candidate selection problem. The problem of suitable tests for computer programming candidates, not blind and not otherwise handicapped, is still being studied as it has not yet been satisfactorily solved either in our country or abroad. A high percentage of candidates succeed in the intake tests but fail while in the study stage of the course. A further percentage succeed in the course but fail in their practical work after placement. From this it is clear that the candidates have certain abilities and proclivities which influence success or failure in the computer programming field, but we have not as yet been able to define these.

It may be assumed theoretically that it is possible for psychologists who deal with these problems to make successful predictions regarding the success or failure of aspirants to the course but there are so far no reliable data available for us to judge whether any progress has in fact been made. Any consideration of the criteria which determine success or failure has up till now been directed at normal and healthy trainees in computer programming and it is obvious that the problem is much more complicated when the trainee is blind, paraplegic, or otherwise handicapped.

In our project we employ the common IBM programming

tests and in addition to these we have developed, in conjunction with the psychological staff of the Hadassah Institute for Counselling, special tests geared to blind computer programmer candidates.

In the first stages of our project we successfully developed special tests for selecting blind candidates for programming and operating Unit Record equipment. All the candidates for whom success was predicted succeeded during their study and were also successful in practical work in their workplace. However, we are still unable to report on the value of the tests we have developed for blind candidates for computer programming.

At this stage when we compare the results of the predictions made by a psychologist at the time of selecting the candidates, the evaluation by the instructors and teachers during the course, and the results of the examinations in each subject taught in this course, we find that there are still some discrepancies in this chain. This is even before we compare the results of the forecast of success or failure in regular work with the earlier predictions. We learn that our evaluation system within the computer programming courses is not efficient enough and we must therefore continue with our efforts and perhaps make changes in it.

Another handicap which prevents us from obtaining a clear picture is the fact that the population tested was small and not sufficiently representative.

To assist in evaluation we have tried to have every candidate pass through a special pre-course in the programming and operating of conventional IBM machines in the first 6-10 weeks before he starts the course in computer programming. In this evaluation period the social workers, psychologists, teachers, and instructors learn to know each individual candidate from all aspects - his attitudes, thinking, logic, behaviour, potential etc. The evaluation results are compared with the data of the previous psychometric and psychological tests made before he entered the program.

We anticipate that the results of this systematic evaluation and comparison with the tests will have a bearing on whether the candidate will continue in the second stage of the course, that is, computer programming, which is the main subject.

We hope that after another two or three courses and further intensive research on these problems, more conclusive tests will be developed which will enable us to predict success or failure in the first stage even before the candidate is accepted for the program, or at least as early as possible during the course.

M E T H O D O F I N S T R U C T I O N

I. FACILITIES AVAILABLE FOR COMPUTER PROGRAMMING TRAINING

A survey of the training facilities in the Computer Programming field in Israel revealed the following. The IBM Education Department receives requests from clients to teach their employees to become computer programmers and accordingly organize courses to fulfil these requirements. A number of public and private schools offer similar courses. The duration of the instructional period varies with the type of equipment on which teaching is done. The curriculum of the IBM 360/20 courses is as follows:

1. Principles of Programming for the IBM 360/20.
   The aim of this course is to teach the beginner programmer in his first stages of ADP (Automatic Data Processing). The course includes an introduction to ADP, general knowledge of the computer, programming the computer storage, and the flow chart.
2. Report Program Generator for IBM 360/20.
   This course includes description of the various components of the unit file organization, punched card utilities, and RPG concerning input-output of card system.
3. Programming the Basic Assembler Language for IBM 360/20.
   In this course the students learn to write programs in

BAL, including IOCS.

II. ON-THE-JOB TRAINING

Sometimes an immediate demand for a small number of computer operators arises. In such an instance there are two possibilities:

1. Generally, when a system changes its equipment from conventional machines to computer 360/20 its highly skilled conventional operators are retrained as computer programmers and its semi-skilled as computer operators. The latter are given basic information about the computer and its functions and are simultaneously trained in the actual operation of the computer.

2. The second possibility is to employ people who do not have any previous experience in ADP, on the basis of an accepted psycho-technical test. The new employee is introduced to the computer room, one of the experienced workers is attached to him, and he learns while executing actual jobs. The new worker studies the literature on the subject (usually the operating procedures) and progresses in the actual work. This method of instruction has been used only in exceptional cases when there was a need for the immediate training of workers for the ADP system. It was

found very difficult and in most cases did not prove successful. This method of instruction can be hazardous for the handicapped because he is less flexible in his ability to move from one position to another, and failure can prove to be traumatic.

III. THE COURSES AT OUR CENTER

Since the preceding instructional methods proved unsuitable for blind trainees, a way was sought to organize suitable courses at the Center.

The experience gained during the initial courses at the Center provided the basis for a study program for the Programming course. The aim of our courses was to teach the students the principles of operation of the conventional IBM equipment, how to write programs for the computer, and how to operate the computer. Both theory and practice were included in the syllabus of the course.

a) General Conditions: The students live together in a hostel where all their needs are provided for. Those who require it receive special social and/or psychological services and from time to time attend evening meetings for group therapy on their current and future problems.

b) Unit Record Preceding Computer Programming: In accordance with the above, and from our experience in

teaching the blind programming, we found that the trainees should be thoroughly familiar with general data processing before studying computer programming. We accordingly found it useful to teach the blind and otherwise handicapped to operate conventional IBM machines and to wire panels for these machines stage by stage according to each of the following machines: Sorter,[x] Interpreter, Reproducing Punch, Collator, Calculating Punch, Tabulator. This section of the program is also utilized as an evaluation instrument.

c) Length of Course: As explained in our Report No. 3, a training program for the blind and otherwise handicapped is difficult to plan within a set time limit. Usually the length of time required for training depends upon the following factors:

1. Planned degree of depth for the study of the subject.
2. Trainee's capacity for comprehension.
3. Instructor's ability to explain clearly.

---

[x]The Sorter is a machine which usually exists even in an installation in which there is a computer. Knowledge of the operation of the Sorter has been found to be of great importance, thus special stress was placed on this machine.

For the blind these factors are of greater importance because they have to compete with sighted people in this profession and accordingly there is a need for them to have the opportunity of gaining maximum practical experience during the time scheduled. Various psychological and orientative problems arise when teaching the blind.

d) Language of Instruction: In Israel, as in other non-English-speaking countries, a fourth factor has to be considered, namely, language difficulties, as most text-books in the IBM field are printed in English, which is not the mother tongue of either student or instructor.

In order to overcome this problem English lessons were included in the syllabus, fitted into the general program in such a way that a lesson covers the material which had just been taught in a theory lesson. The Systems Reference Library also served as textbooks for English. Accordingly, this method has a twofold purpose: a) study of the professional language and terminology, and b) review of material studied. Hence it is important that the English teacher has a good knowledge of IBM work as well.

e) Course Split for Trainees' Evaluation: The course is divided into two parts, Unit Record and Computer Programming,

making it possible for a check to be kept on the progress of the trainees. The period between the two parts gives both the instructor and the student sufficient time and experience to evaluate independently the student's aptitude for and compatability with this work before commencement of the programming course.

After completion of Part 1 a detailed discussion on each student is held to determine the extent of his grasp of the subject, his adjustment to the work and his ability to progress to the second stage of the studies, which cover the computer itself. This ensures maximum success in work for those who complete the course, while those not suited do not continue their studies in this field.

f) Detailed Syllabus and Length of Study: The study of the machines in accordance with the following daily program requires a period of twelve months, covering five study days a week. The period allotted for the study of the different subjects is as follows:

PART 1

1.	Basic data processing	1 week
2.	Sorters 082, 083	2 weeks
3.	Interpreter 557	1 week
4.	Collators 077, 088	2 weeks
5.	Reproducing Punch 519	1 week
	Total	2 months

PART 2

1.	Introduction to Computer Programming	1 week
2.	Programming the Punched-Card Utilities	3 months
3.	Programming the Report Program Generator	3 months
4.	Programming the Basic Assembler Language	4 months
	Total	10 months
	Grand Total	12 months

g) Daily Schedule: The daily study program for this course is planned as follows:

08.00 - 10.00	Checking of Programs[x]
10.30 - 12.00	Theory of Programming
13.00 - 14.30	Checking of Programs
14.45 - 16.30	Theory of Programming
16.45 - 18.15	Monday, Wednesday: Mathematics Sunday, Tuesday, Thursday: English
19.00 - 20.30	Machine Typewriting[xx]

---

[x] "Checking of Programs" means processing of programmes on the computer.

[xx] This subject is dealt with when dealing with the problems of Input. See page 27 .

In contrast to the usual training courses for sighted pupils, the Center's course provides practical as well as theoretical instruction.

In general, the daily schedule is built on the principle of maximizing the trainees absorption of the studies. The assigning of work to be done overnight on their own time maintains a steady level of concentration and obviates the need for a "warming up" period each morning, which wastes time and is not efficacious from a pedagogical viewpoint.

The four daily lessons which are directly related to Programming are to be taught by two instructors, one for theoretical instruction and the other for review and practical application. By alternating the instructors and the corresponding subject material, a higher level of interest is achieved.

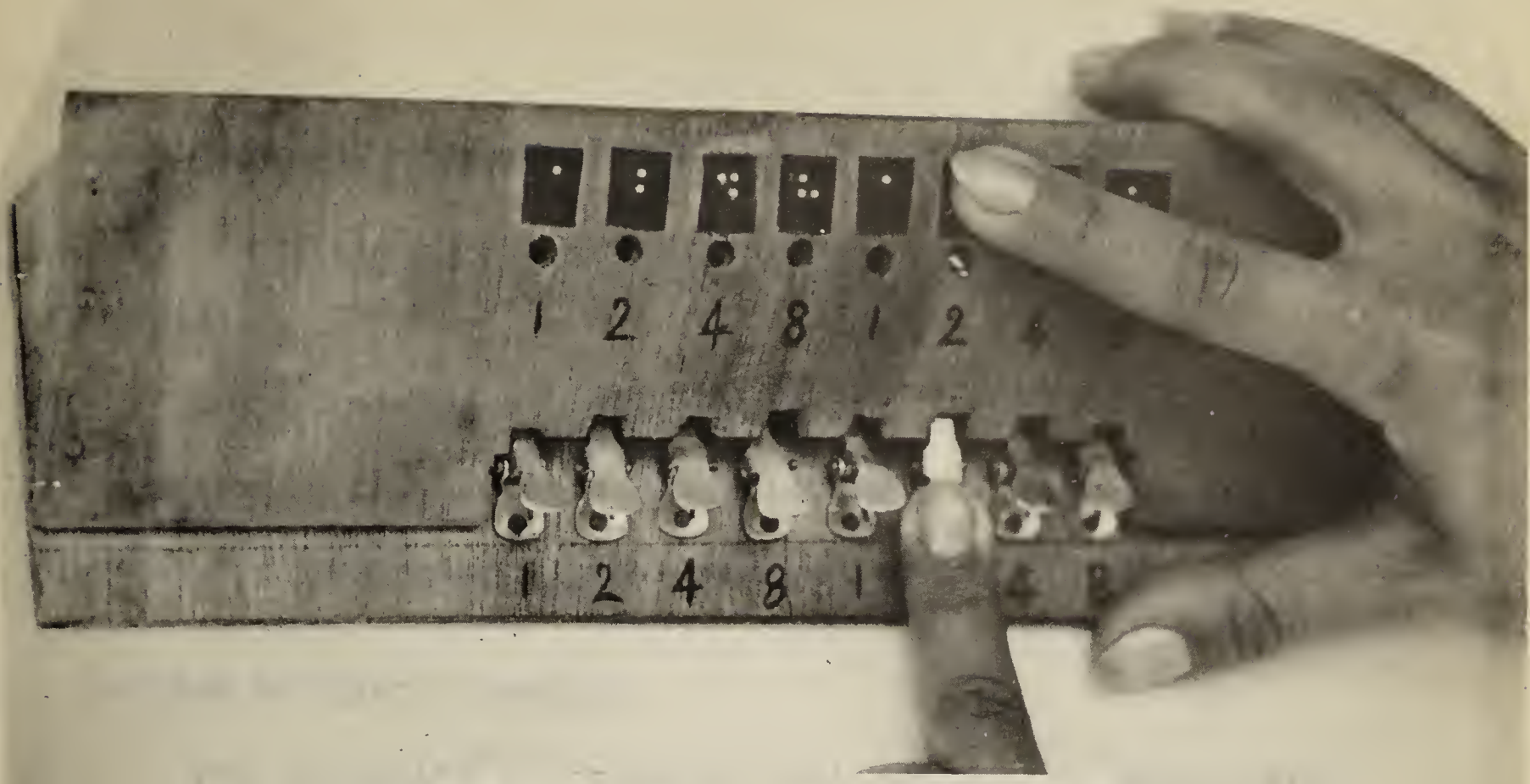
The daily program assures that the instructors will be able to confer with one another between lessons and coordinate the various aspects to a much greater level of integration.

These are the factors which were considered when designing the daily program to be followed after the first week of introductory material.

h) Didactics of Studies: The syllabus of the theory lessons of the course for the blind and otherwise handicapped has been adapted from the material used for the teaching of theory in IBM Education Department publications and other handbooks elaborated for the purpose of teaching handicapped pupils. Special care has been taken for the blind trainees; special multi-sensory aids have been designed and developed for them but as the blind pupils in our course have both sufficient background on ADP and sufficient experience in working on IBM machines by the time they reach the stage of studying the programming of the computer, they have enough familiarity with the subject to be able to understand the method used in spite of their lack of sight.

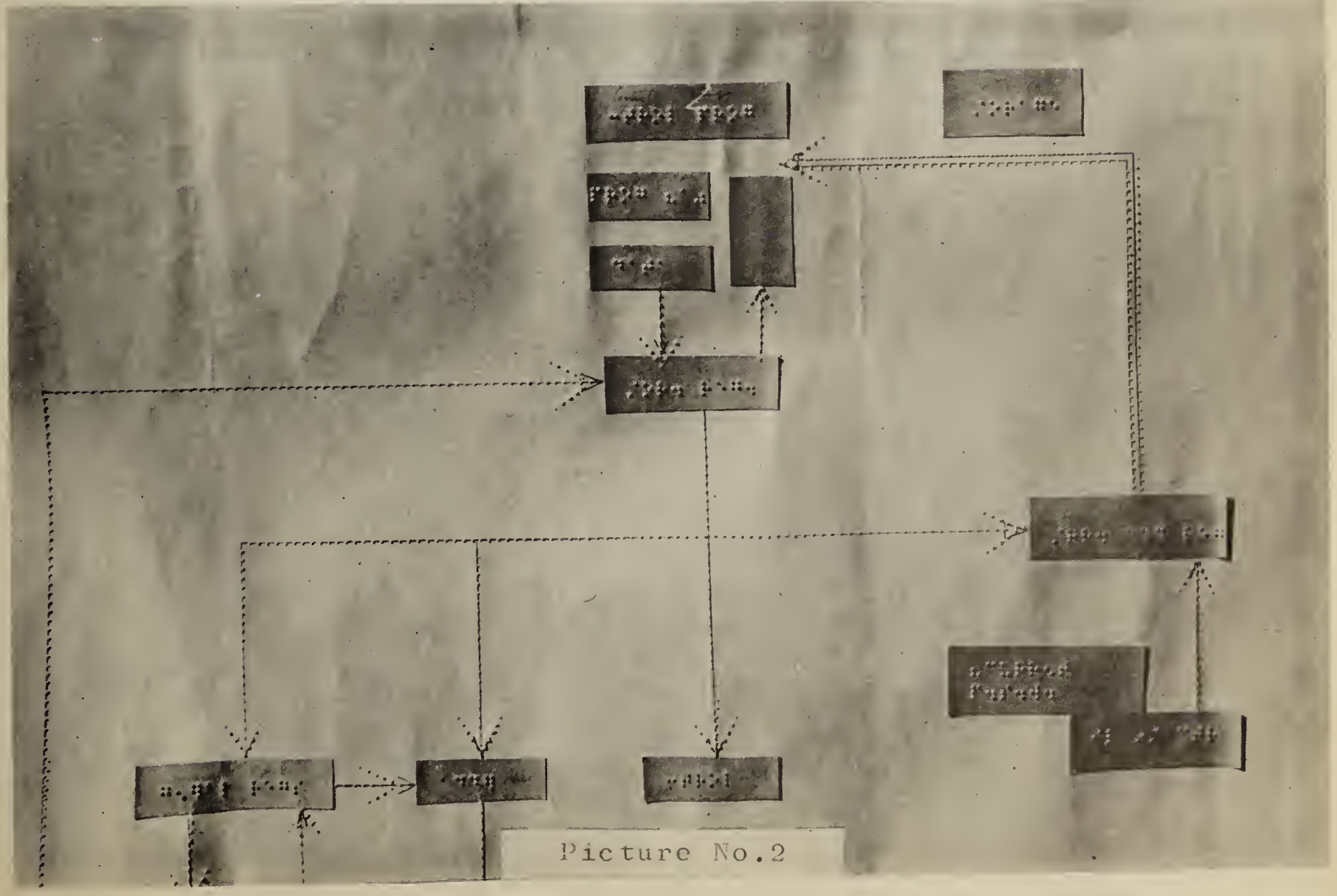
One of the multi-sensory aids which have been designed for the comprehension of the lessons is shown in picture No. 1 and another in picture No. 2 . Additional means have also been used, as described in Report No. 3,

Our experience has also proved that theory lessons can be given successfully to the blind and otherwise handicapped together in one class. This is, of course, on condition that the instruction is given in a small class, within a rehabilitation setting, on a more or less individual basis,



Picture No.1

Explanation of the byte.



Picture No.2

MACHINE INSTRUCTION FLOW

using special techniques for each kind of handicap, and that, as well, theory and practice are coupled as already specified.

The blind use a braille typewriter and a tape recorder to take notes and to clarify and revise these notes, if necessary, upon their review.

IV. OUTLINE OF THEORY LESSONS

The following are general outlines of the three parts of the course, the PCU, the RPG and the BAL. Teaching material from the IBM Education Department and the actual experience of the Center's staff were the basis of the above.

GENERAL OUTLINE FOR COURSE IN PCU
FOR 360/20

I Introduction to Punched Cards Utilities

A. General discussion of Utility Programs

B. Functions of PCU

C. Functions of PCIC

D. Function of Program Deck

II COLLATE

A. Functions

B. Cards involved

1) COLAT used for matching and/or merging only.

2) FINDR used for selection, either relative to constant or within the same card.

3) PLACE used for pulling or inserting cards by a numeric factor, or first or last card of groups.

C. Preparing Program

1) Analysis of problem.

2) Analysis of data.

3) Documentation.

4) Actual coding.

5) Insertion of programmer's cards in deck.

6) Operation and Diagnostics.

7) Messages.

III COLAT card

A. Functions performed by this card.

B. Detail information of card by columns.

C. Exercises with the COLAT card only.

IV FINDR card

A. Functions performed by this card.

B. Program Logic Flow

1) Decision Tables.

2) Flow-charts.

C. Detail information of card by columns.

D. Exercises with FINDR card only.

E. Exercises with FINDR and COLAT cards.

V PLACE card

A. Functions performed by place card.

B. Detail information by columns

C. Exercises with place card only.

D. Exercises with COLAT, FINDR and PLACE cards.

VI Summary of Collate Program

A. Functions performed by each card.

B. Developing of keys for the trainees' use (See IBM manual appendix).

VII Sort Utility Program

A. Functions

B. Advantages of Computer/Conventional Sorting

C. Sort parameter cards.

VIII Gangpunch/Reproduce Utility Program

A. Functions of Program and conventional machines replaced.

B. Information Sources

1) Data cards

a) Group 1 (types A-C) input to Reproduce.

b) Group 2 (types D & E) input to Gangpunch.

c) Group 3 (types F-H) output from all operations.

2) Constants.

3) Counter Information.

C. Program Cards.

1) Control Card

2) Selector Card

3) Field Definition Card

D. Detail of Control Card for Gangpunch Operations

E. Detail of Selector Card for Gangpunch Operations

F. Detail of Field Definition for Gangpunch Operations

G. Developing concept of Object language

1) Source/Machine Languages

2) Advantages of object deck

GENERAL OUTLINE FOR COURSE IN RPG FOR 360/20

I RPG - General Introduction

A. The nature of the program

B. Common functions of RPG

1) Known functions (common to RPG & CPU).

2) New functions and associated terms

C. Steps in utilizing RPG

II Writing a program in RPG

A. Relations between CPU, RPG and BAL

B. Terms of specific significance in RPG - definition and examples

1) File

2) I/O

3) Record

4) Names

5) Instruction

C. Specification sheets and cards

III Introductory Program Examples

A. Problem analysis

1) Given data

2) Desired results

3) Processing required

B. Solution

1) Flow of information referring to problem analysis.

2) Detailed description of determination of processing by specification.

IV Indicators - as a means of providing for common problems in programming

A. Conditioning of processing

B. Control level and control break

C. Matching

D. Terminating a job

V RPG Program Logic

VI Introductory remarks on specification procedure

A. Mandatory and optional specifications

B. Fields common to all specification forms

C. Notes on the organization and use of IBM SRL publication

VII File description specifications

A. Introduction

B. Machine units and features

C. Specification sheet entries

D. Examples of specifications

E. Exercises

VIII Input specifications

A. Introduction

B. File and card-type identification

C. Field description

D. Examples of specifications

E. Summary and relations to file description specifications

F. Exercises

IX Calculation Specifications

A. Introduction

B. Kinds of calculations provided by RPG

C. Testing results of calculations

D. Operations

1) Arithmetic

2) Move

3) Compare and zone-testing

4) Setting indicators

5) Branching

E. External routines

F. Summary and Exercises

X Tables

A. Use of tables

B. Lookup operation

GENERAL OUTLINE FOR COURSE IN BASIC ASSEMBLER LANGUAGE FOR 360/20

I Binary and Hexadecimal Arithmetic

A. Recognizing configurations and numerical values.

B. Appreciating concept of Bits and Bytes

II Instruction Formats

A. SS format

1) Character instructions one length

2) Arithmetic instructions two lengths

3) Discussion of packed format

B. SI format

1) Concept of masks

2) Differentiation between masks and constant-literals

C. RS format

Concept of registers

D. RX format

Concept of Indexing

III Addressing

A. Base Registers

B. Concept of Displacement

C. Labels as codes for Base/Displacement Addresses

IV Instructions

A. Executable Instructions

1) 1/O instructions

a) Logical IOCS and Macros

b) Programmed 1/O control

2) Regular Instructions

B. Compiler Instructions

1) ORG

2) Use of

V Summary of All Non-1/O instructions

VI Indexing and Loop control

VII Area Definitions

a) DS

b) DC

VIII Programmed 1/O Control

a) XIO

b) CIO

c) TIOB

IX Logical IOCS

a) Use of Macros

b) File Definitions

c) Parameters for DTF's

X Relationship between DTF and Macros

XI Summary

PROBLEMS OF THE INPUT

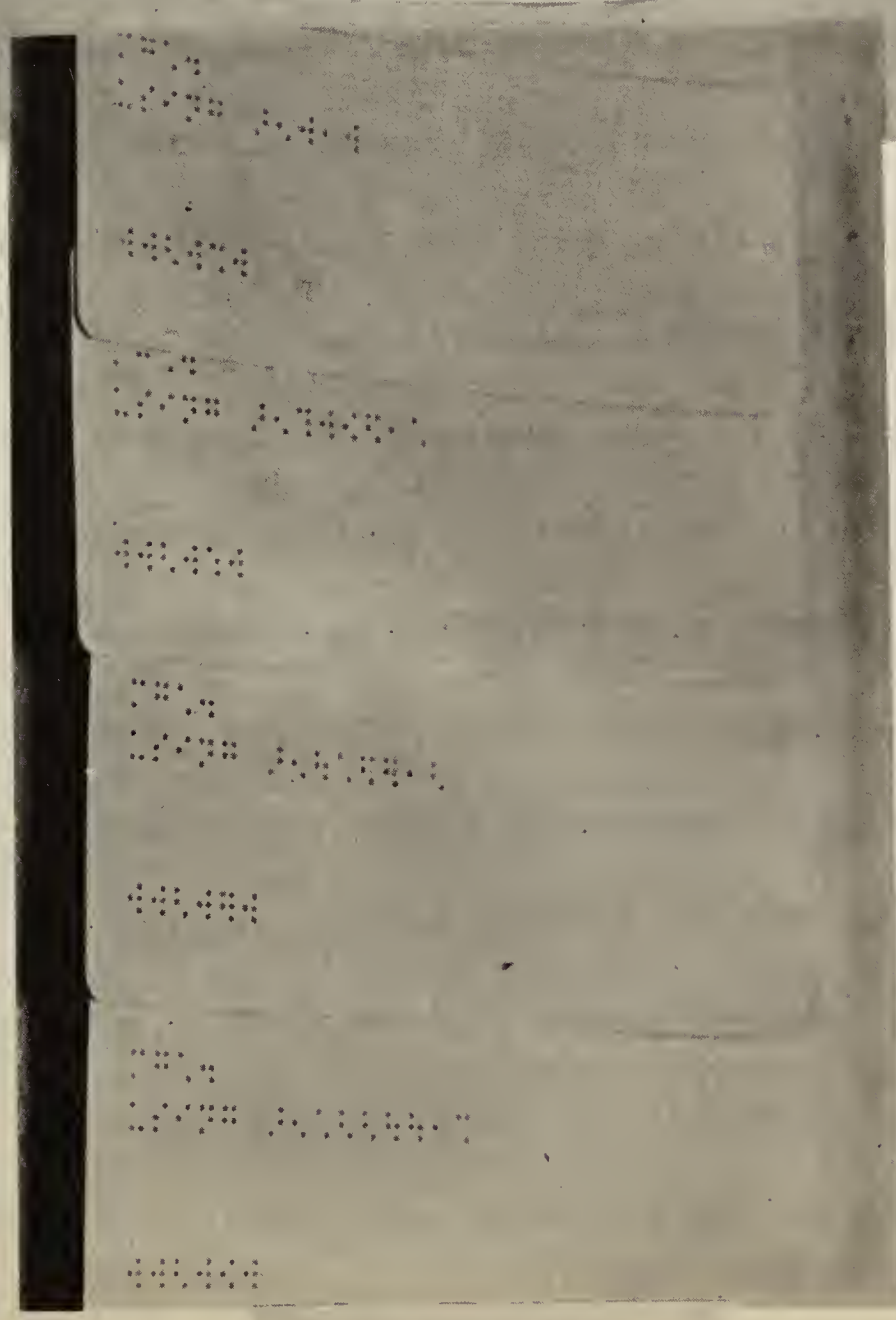
One of the more complicated stages of programming is the input. This stage can be broken down into several operations, as follows:

FLOW CHART: When writing a program, the programmer describes the outline of the program on the basis of the flow-chart. This method of description enables him to see a whole program in one glance. Although the blind programmer cannot use this method, he has been found able to concentrate on a complete small-to-medium program without using any method of description. However, for a medium-to-large program (and sometimes for small programs as well) the blind have to use another method of description, of which two have been found suitable and are illustrated here.

Picture No. 3.: The page is divided into three columns. The first is for the sequence numbers, the center for the contents and the last for the reference loop.

Picture No. 4 : The orders are written on IBM cards or pieces of paper containing sequence order. The blind can arrange these cards as required.

Picture No.3



Picture No.4

CODING: There are generally standard IBM forms for each programming language which must be filled out according to the program. As described on page 11, our daily program includes machine typewriting, knowledge of which enables the blind programmer to type his own programs for the key-punch operator. For the blind, we designed forms sized specially for the typewriter. One corner of the form was cut so that the blind could orient themselves to the direction of the page. The headings and columns were already printed on the page and their spacing was synchronized with that of the typewriter so that when the page was inserted into the typewriter the blind could know their location. It was found, however, that the use of these special forms took too much time and occasionally lacked precision so that sometimes the selected column was not typed in.

Another means of coding is to have the blind programmer use a blank page, on which he types the numbers of the columns as headings and then types the contents accordingly. By using this method the blind programmer can be sure that he types in the proper column and he is not limited as with the previous form. This method saves keypunch time as the information is more concentrated on the page as compared with the IBM forms, which have the information in different areas.

This process does not take more time than that needed by a sighted programmer to code the same program using the IBM forms.

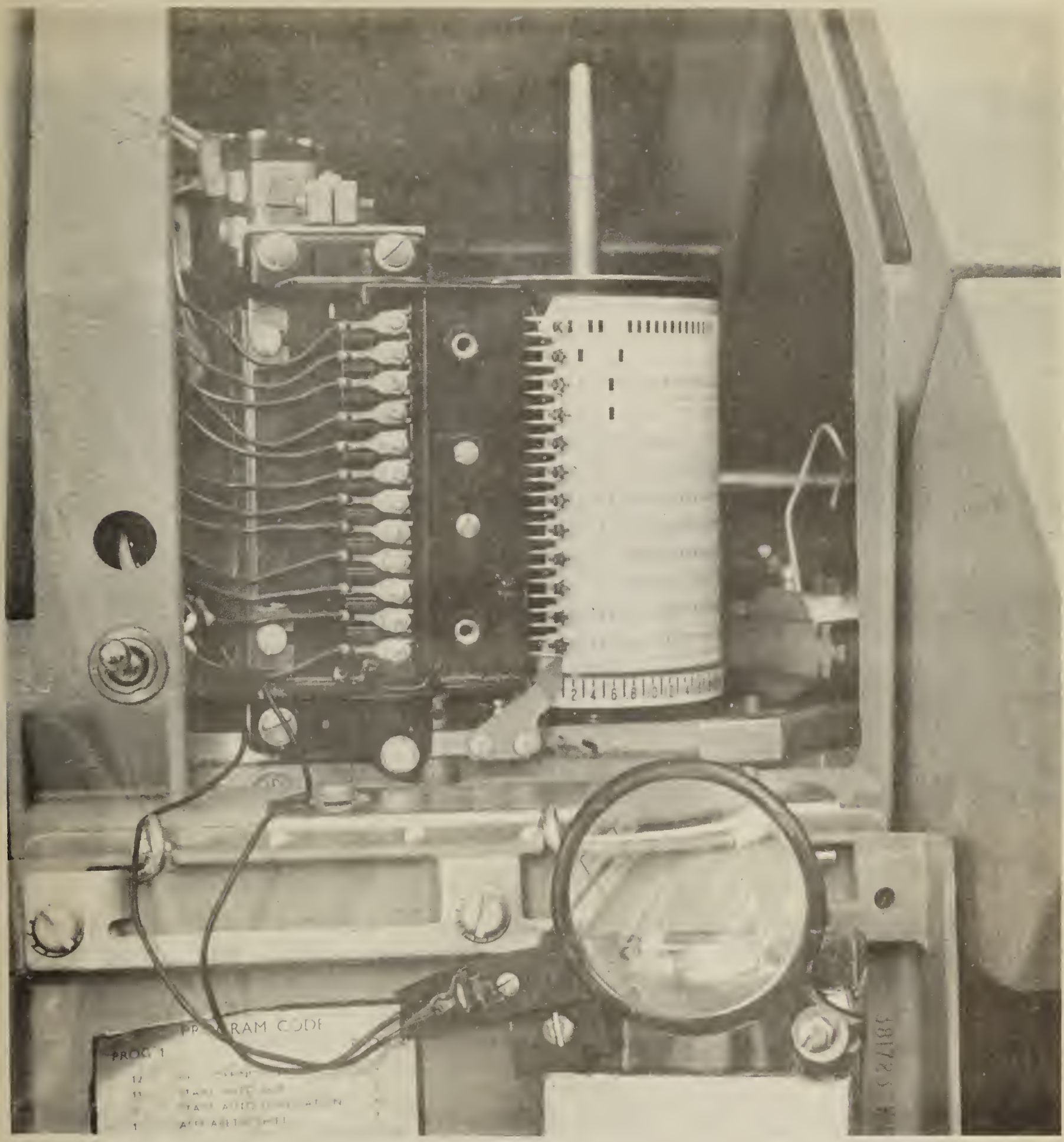
To overcome these problems we are presently testing, in cooperation with the National Physical Laboratory of Israel, an electronic adaptation of an ordinary typewriter which embosses a paper tape with the braille equivalent of the typed page. Thus when the programmer types his program in the usual manner he receives simultaneously a braille transcript of it, which he can check immediately for typing errors.

This instrument has proved successful in intensive use over a five-month period. The machine has helped give the programmer self-confidence in his communication with the key-punch operator. See Picture No. 5

In some installations the programmer is able to punch his own program. This he is sometimes obliged to do because of specific problems, e.g. scientific programs with complicated terminology for the keypunch operator, etc. Having the blind punch his own program was found to be the most concise and productive way. Punching takes the same time as typing and not more mistakes are made, so that the time spent in both typing and punching is reduced by 50 per cent. The partially sighted can punch his own program and

can check the leader card with a special magnifying glass and special light (See Picture No. 6). The same method can be adapted for the purpose of reading the interpretation on the punched card. The totally blind can check punched cards with the card reader instrument (see Picture No. 7), but this takes too much time if compared with a sighted programmer, and we are now designing an electronic instrument that will read punched cards and transcribe their contents into braille at a high speed.

Picture No. 5. A regular typewriter has been adapted electrically to operate a Braille paper embosser. The output is a typed page and, simultaneously, a braille paper tape.



Picture No. 6 : A small lamp has been connected to the drum card leader and a magnifying glass has been connected accordingly on an axis. When a partially sighted uses this instrument he is able to turn the magnifying glass on the axis and see the card leader through it.

Picture No. 7 : This model is a combination of both instruments: the light-identifier and the card-reading instrument. The result is a heavier instrument which can remain fixed, thus providing additional stability. This instrument consists of electronic components. It has horizontal and vertical braille scales and can be moved from column to column by turning a knob or by sliding it freely along the scales after operating a release-lever. The latter lever is capable of regulation by the operator either to intensify the click or to meet expansion vagaries caused by the weather.

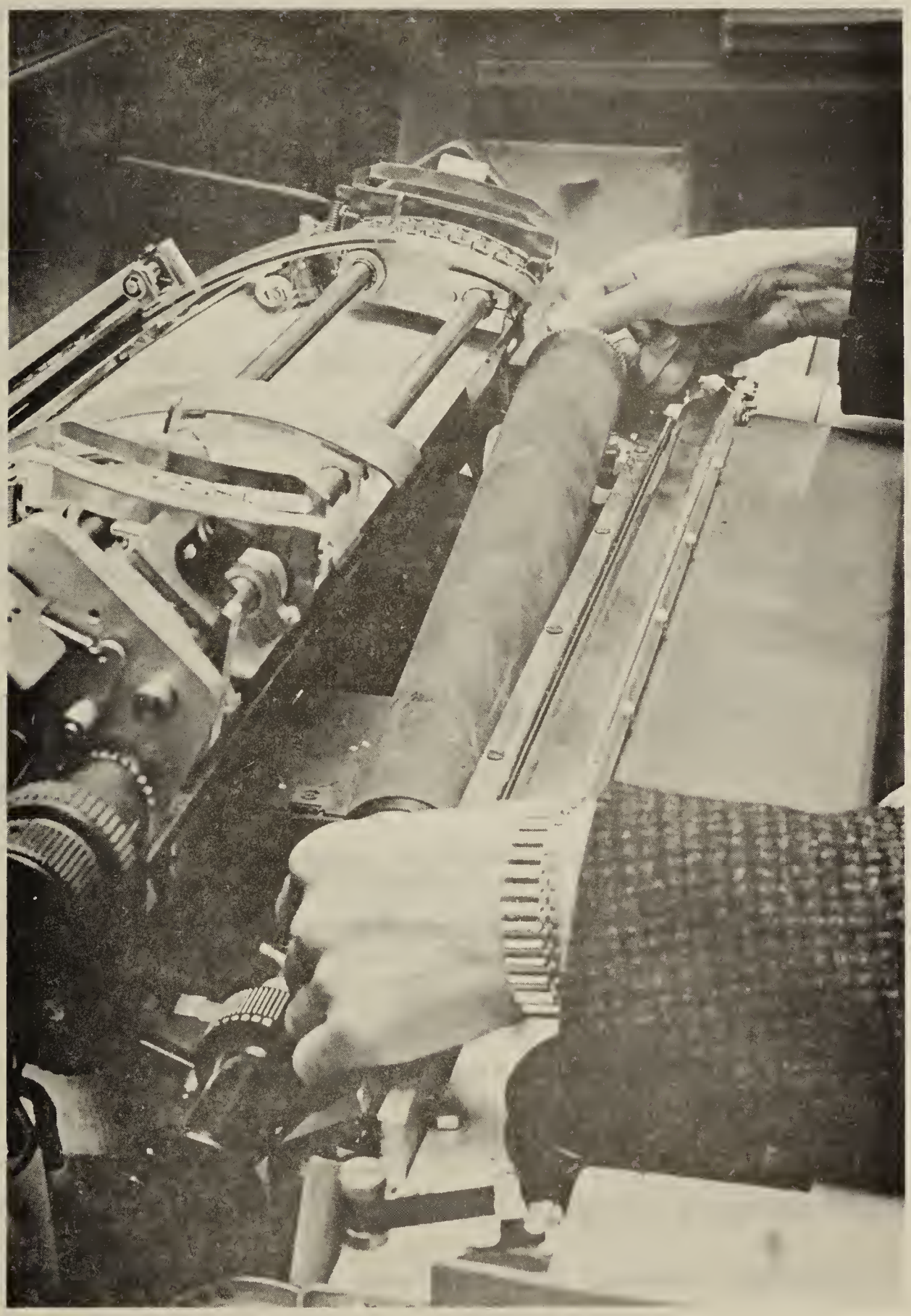
PROBLEMS OF THE OUTPUT

The most complicated stage for the blind programmer is the checking of output material. The method employed for this purpose uses the dot of the chain of the printer with a special program for braille transcription which serves to transcribe the regular block letters to the braille combinations. This method of braille compilation has been in use in different places for some years. It uses twelve times more paper and takes approximately ten times more computer time, which is both expensive and which produces an immense quantity of braille printed paper that sometimes interferes with the debugging.

The IBM Company has developed a special device which enables the 1403 printer to print raised braille. This can only serve installations which use medium and large sized computers containing the 1403 printer, providing that the rental, which is $2,000 per year, is not prohibitive. This method of printing braille produces well raised dots and its speed is usually equal to the regular method. The device prints a maximum of 366 braille lines per minute with the IBM's 1403 N1 Printer and 200 braille lines with the 1403 Model II Printer, each line containing 52 figures. This

means that each block letter line is broken into three braille lines (132 type-bar positions divided into 52 braille figures). 366 braille lines are equal to 122 block letter lines but these 122 block letter lines could be produced by the same printer whose speed is 1100 lines per minute, in one-ninth of a minute. Comparison of braille with regular printing shows that braille printing with this method takes nine times longer than regular printing, wherever the computer time by dots is about ten times more than regular printing.

Whatever the advantages of the special device of IBM, it is limited in that it can only be used with the 1403 printer. It cannot be regarded as a general solution. In any case, to solve the problems connected with the 360/20, for which the regular printer is the 2203, a different method for printing braille had to be developed. This we have done within the framework of our research. The high quality of braille print has been achieved by removing the printer's roller of the 2203 (which can be done very easily) and replacing it with a reserve roller covered with soft rubber (see Picture No. 8).



Picture No.8

Regarding the braille printing itself, a translation program has been constructed which enables the blind programmer to debug and compile his programs himself, be it written in RPG or BAL.

To print braille the carbon ribbon is removed and the printer types braille on regular continuous pages, with no copies, as opposed to other methods. For each programming language there is a special braille program so that when a blind programmer wants to compile and debug his program he has only to insert this special braille program. However, as the IBM regular programs are constantly being changed and improved, the special braille programs have to be changed accordingly.

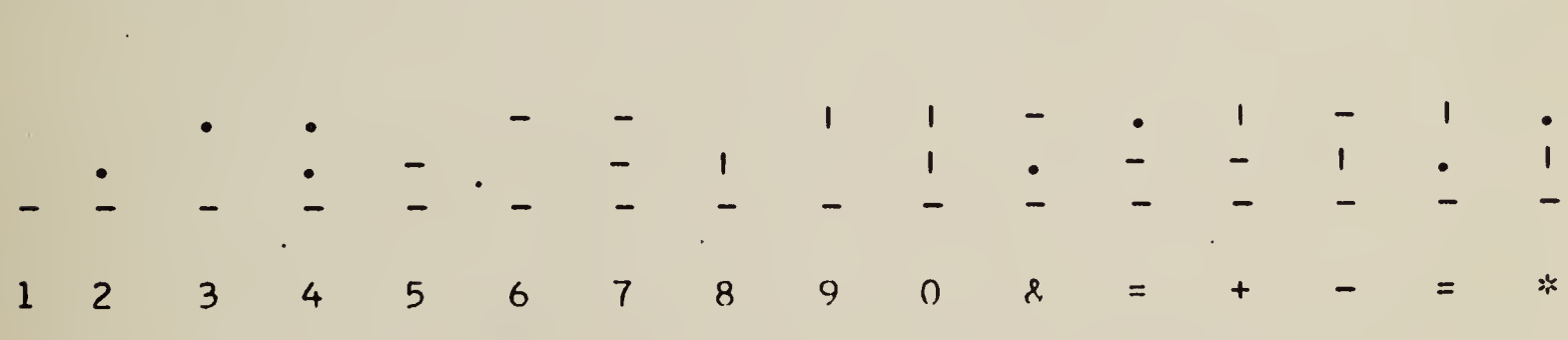
FUTURE PLANS

While the solutions evolved serve us well at present, we feel that they are not yet final. In future solutions will be sought which can be applied in most computer installations. This means that our efforts will be applied in the following directions:

1) The production of a special monitor program which we hope will be a general one for the common computer languages, such as PCU, RPG, BAL, etc.

2) The designing of a special braille chain or ruler which will reduce the computer time to one-third of its present length when printing braille.

3) The developing of a new raised figure combination system which will reduce space and computer time because of its use of three kinds of figures which, in various combinations, give 48 configurations. As shown hereunder, it can contain most of the configurations needed for programming.



4) We think that the electronic instrument shown on page 29 can be extended to include also punch-card inlet and will be able to transcribe its contents into braille. This new instrument will then serve as off-line equipment. The blind can insert the punched cards received during the debugging of the programs into this instrument and receive the output in braille without any extra load on the computer's printer.

JOB BREAKDOWN

The following job breakdown has been thoroughly elaborated and analyzed for the purpose of teaching the operation of the computer to the partially sighted and to otherwise handicapped who are seated in wheelchairs. It has been done mainly for the paraplegic, but as we teach partially sighted as well as otherwise handicapped, we thought that efforts should be made to teach all the trainees how to operate the computer, not as regular job operators but for compiling their own programs for debugging.

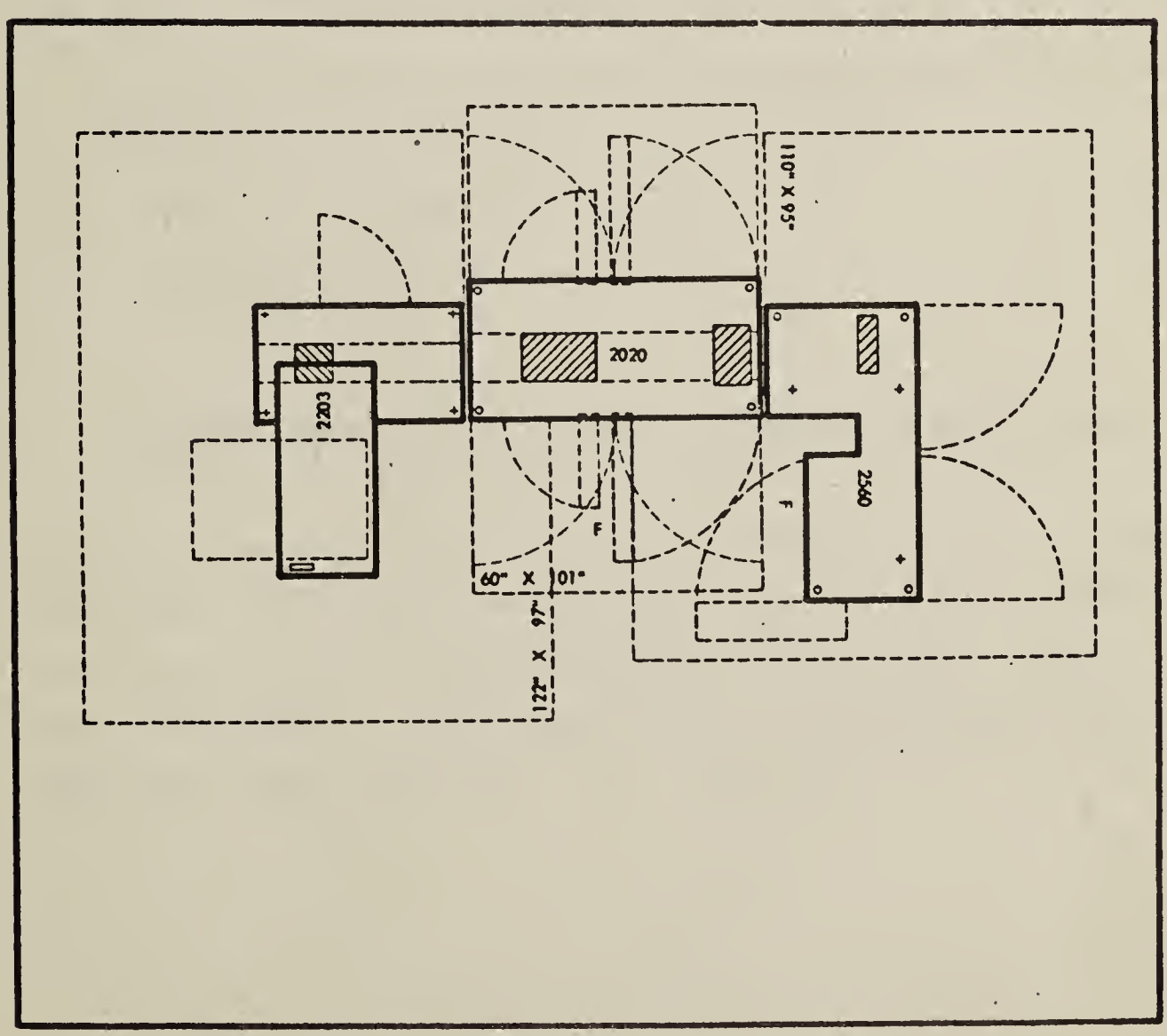
It appears that, as opposed to the fairly large number of operations in which vision is of assistance in the performance of their work, there are only a few operations in which vision is indispensible. Regarding amputees, paraplegics, etc., who are seated in wheelchairs, the job breakdown shows that wherever the performance of the job requires the operator to stand, they are able to perform the job seated in standard wheelchairs.

Two of the following pictures show how the most complicated jobs are executed while seated. If the wheelchair can be raised and lowered, the effort of operating a computer would be much reduced. An important consideration is that there should be sufficient room to move around the computer - as shown in the following sketch.

The only change in the working conditions that has to be made is in the transportation of a full box of continuous paper, which is heavy even for a non-handicapped man, and for this purpose we built a special low carriage. Although this carriage appears small, it in fact accommodates all kinds of continuous paper boxes, does not take up too much room, and is stable. We removed the lower paper tray so that the handicapped seated in a wheelchair can push the carriage from the storage area to beneath the printer. This system of transportation has also been found to be of benefit to a non-handicapped colleague.

IBM System/360 Model 20

Space required for installing IBM 360/20 Computer as recommended by IBM Engineering Department. It allows convenient mobility for a wheelchair-seated operator.



Scale 1:50 קנה מדה

PREPARE THE COMPUTER

1) Remove the cards of the MFCM, the CPU, and the printer.

2) Check the hoppers to be empty of cards.

3) Press "power-on" button in the CPU.

4) " "npro" " " " MFCM.

REMOVE CARDS FROM STACKERS

1) Hold a group of cards in one hand.

2) Pull the cards towards you.

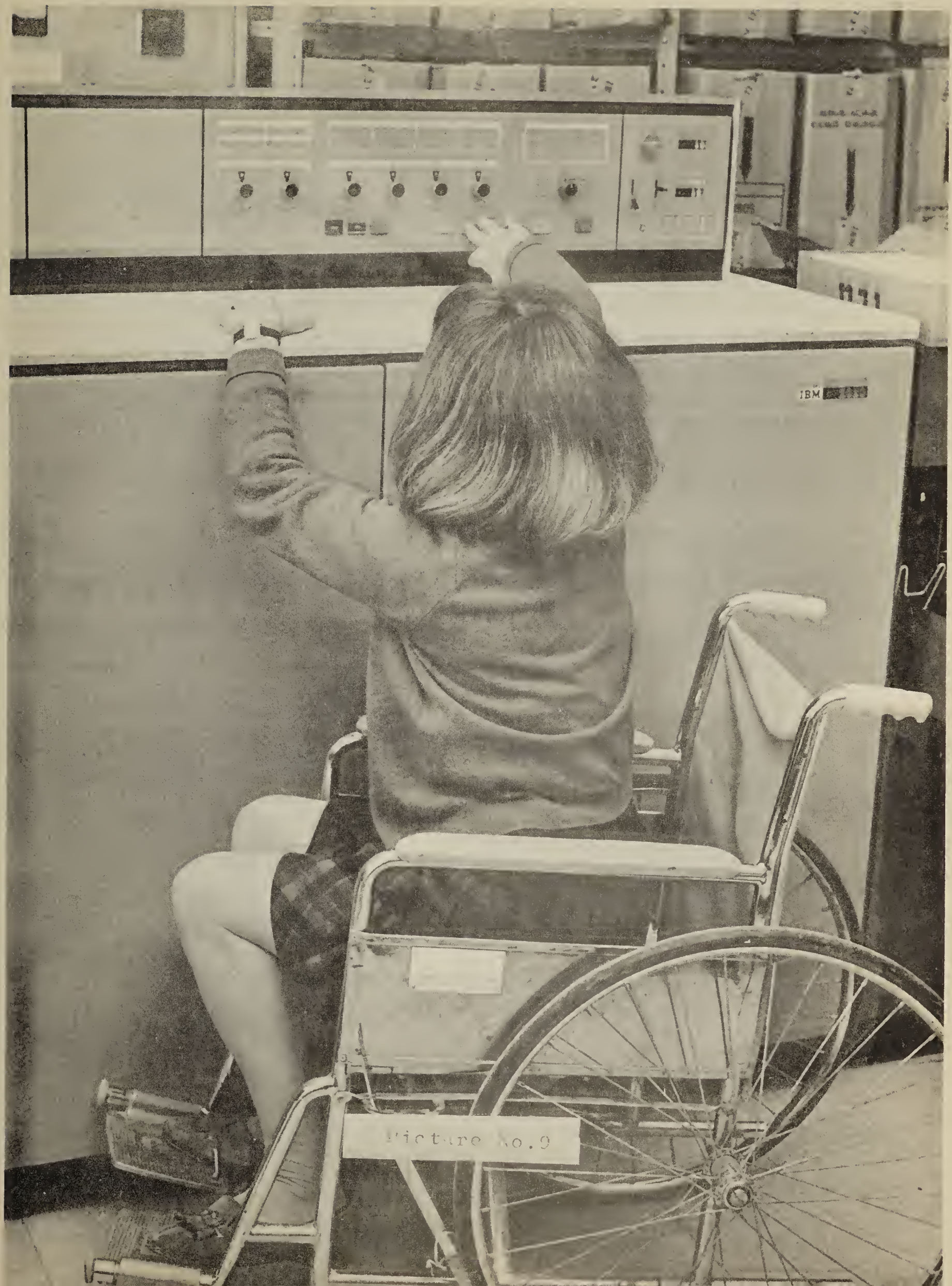
3) Remove cards from stacker by allowing the spring holder to go back to its station.

4) Joggle and straighten the cards.

5) Insert cards into the box.

SYSTEM PREPARATION AND OPERATING PROCEDURES

These jobs are specifically described in the manuals dealing with the Function and Operating Procedures (especially Form C26-3800-0, pp. 11-14). The following picture (no. 9) shows how a wheel-chair-seated operator can reach the CPU's panel.



Picture No. 9

PRINTER SET-UP

1) REMOVE THE CONTINUOUS FORMS PRINTING.

   a) Open the cover.

   b) Pull handle of printer's press.

   c) Push down handle of printer's press.

   d) Release spring pull of printer's handle. (The spring inserts the pin into the lower hole).

   e) Turn leftward press knob (until the end).

   f) " rightward "in/out" knob towards out.

   g) Raise up two paper holders (outwards from you).

   h) Pull rightward and leftward two teeth holders.

   i) " the type-bars unit.

   j) Raise up the paper to release from teeth (from both sides).

   k) Pull the paper from the slider towards you.

   l) Release the paper towards the type-bars.

   m) Collect the released paper on the package following the folds.

   n) Pull down two paper holders.

   o) Take rest of paper to storage.

2) BRING PAPER.

a) Go to storage.

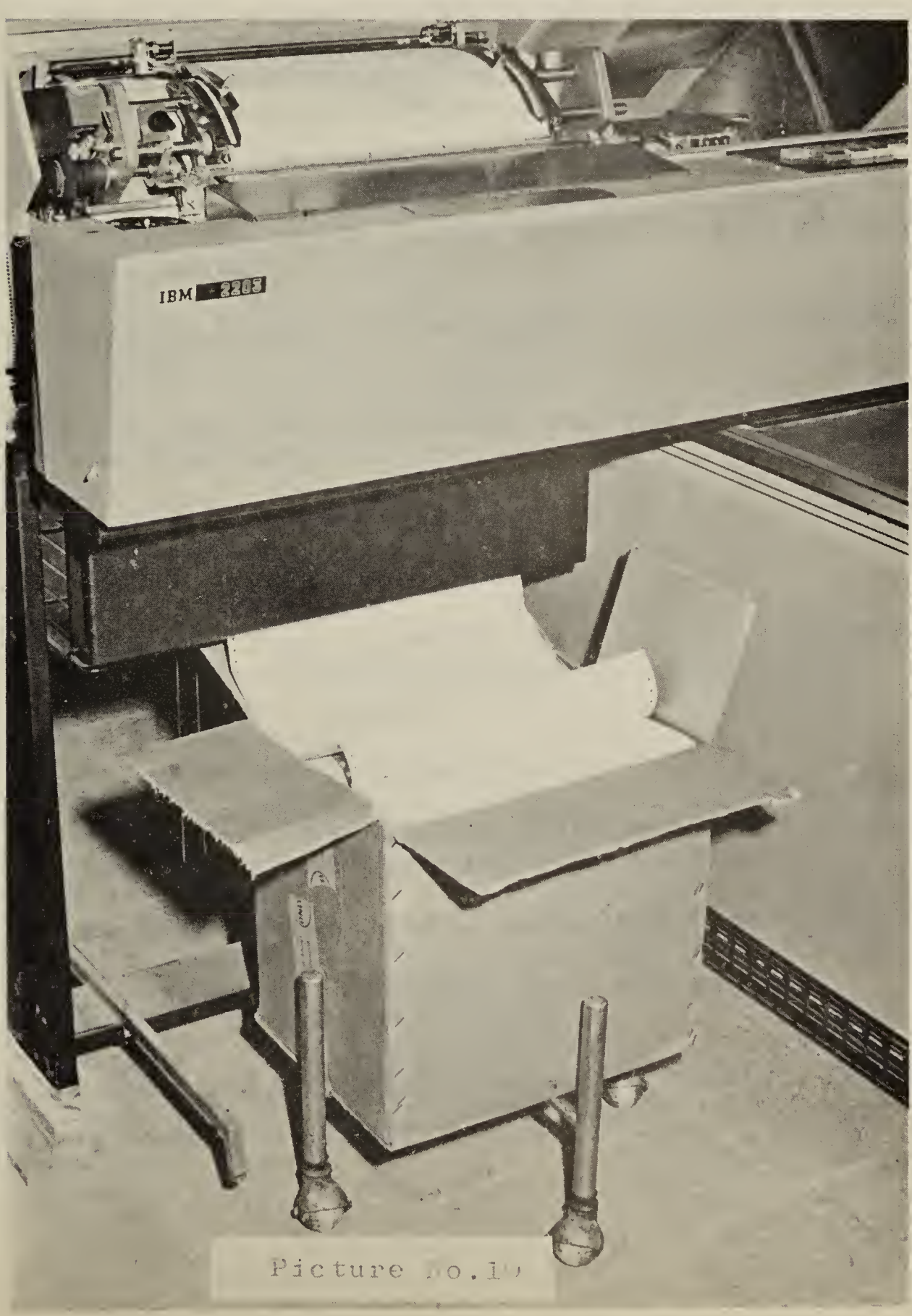
b) Chose the suitable paper.

c) Check the paper (quality and quantity).

d) Bring the paper to the computer.

e) Put the paper under the printer.

f) Make sure that the carbon paper is outward from you.



Picture No.10

3) LOAD THE PAPER.

a) Hold (in left hand) the right corner of the paper.

b) Insert the paper from left side of the printer (in the special place).

c) Bring the paper to the center of the printer.

d) Press on the handles that release the holder of the paper on both sides.

e) Raise up the two holders of the teeth.

f) Insert the teeth in the holes of the paper on both sides.

g) Pull down two holders of the teeth to loosen it up.

h) Move and steer the paper holders according to the type-bar.

i) Pull towards you the handles of the paper holders.

j) Push down the paper holders.

k) Turn the right knob of the carriage rightward until the type-bars will stay against the chosen line.

l) Put back the type-bar.

m) Pull the press handle, raise and loosen (according to the table on the printer).

n) Turn the press handle until the chosen mark.

o) Turn leftward "in/out" knob towards in.

The following picture no. 11 shows how a wheel-chair-seated operator is able to perform this job.

Picture No.11

4) CHANGE OF V.F.U. PAPER STRIP.

a) Raise up brush holder.

b) Turn to the left down knob.

c) Raise up stretch knob to release paper strip.

d) Remove paper strip.

e) Hold chosen paper strip with 12 towards you and 1 towards the computer.

f) Insert a new paper strip under stretch knob.

g) Raise up stretch knob.

h) Put the holes of the paper strip on the chain teeth of the upper wheel.

i) Release stretch knob.

j) Press lightly downward stretch knob and close it by turning to the right.

k) Push to the left the brush container until a lock-click is heard.

1) Make sure that the holes are set on the teeth.

5) CHANGE TYPE-BAR UNIT.

a) Turn leftward the type-bar guide knob until the edge can be seen on the right.

b) Hold the edge of the type-bar and pull it straight.

c) Return the type-bar into its pocket.

d) Remove the chosen type-bar from its pocket.

e) Insert the chosen type bar into the tunnel when the cone edge is on the left.

f) Push it straight ahead until the end.

g) Turn rightward the side knob of the type-bar until the mark.

Note:

1) Turn off printer's cover.

2) Remove all cards and other material of the back of the printed continuous paper.

3) Push down "reset" knob.

4) Push down "carriage store" knob.

CHANGE OF CARBON STRIP IN THE PRINTER

1) REMOVE CARBON STRIP.

   a) Remove continuous paper.

   b) Remove type-bar unit.

   c) Raise up left carbon strip roller.

   d) Release carbon strip from both left guides.

   e) Raise up carbon strip along type-bar unit.

   f) Release carbon strip from both right guides.

   g) Raise up right carbon strip roller.

   h) Throw one of the carboh strip rollers into the basket.

   i) Insert pencil into the central hole of the other roller.

   j) Hold the pencil as an axis and pull with the second hand the carbon strip into the basket until it empties.

   k) Release end of strip from the pin holder of the roller.

2) LOAD CARBON STRIP IN THB PRINTER.

a) Insert the new roller on the right axis (as described on the printer).

b) Put the ribbon along the right guides.

c) Put the ribbon along the print area.

d) Put the ribbon along the left guides.

e) Pinch the edge of the ribbon with the pin of the empty roller (as described on the printer).

f) Put the roller on the left axis.

g) Check the ribbon against the description on the printer.

INPUT OF CARDS

1) Remove cards weights.

2) Joggle and straighten cards (on the MFCM's desk).

3) Put cards face down, 9 ahead.

4) Put back cards weight on the cards.

5) Push down "start" key of the MFCM.

6) Push down "load" key of the CPU..

REMOVE CHIPS FROM THE MFCM

1) Stand behind the MFCM.

2) Remove all cards and papers from MFCM's upper covers.

3) Raise up right upper cover.

4) Raise up left upper cover.

5) Open right cover to the right.

6) Open left cover to the left.

7) Raise up the handle of chips box.

8) Pull up chips box.

9) Throw away chips from box.

10) Put back chips box.

11) Close up left door.

12) Close up right door until it locks.

13) Bring down left upper cover.

14) Bring down right upper cover until it locks.

ADJUST INTERPRETING LINES

1) Stand behind the MFCM.

2) Remove all cards and papers from MFCM's upper covers.

3) Raise up right upper cover.

4) Raise up left upper cover.

5) Open right cover to the right.

6) Open left cover to the left.

7) Bring down clutch lock handle.

8) Adjust interpreting lines as described in the program.

9) Raise up clutch lock handle.

10) Close up left door.

11) Close up right door until it locks.

12) Bring down left upper cover.

13) Bring down right upper cover until it locks.

CHANGE CARBON RIBBON IN THE MFCM

1) Stand behind the MFCM.

2) Remove all cards and papers from MFCM's upper covers.

3) Raise up right upper cover.

4) Turn leftward (open) right holder screw until the hinge releases.

5) Push rightward the hinge and close the holder screw.

6) Turn leftward (open) left holder screw until the hinge releases.

7) Remove the hinge from its axis.

8) Push the hinge leftward and close the holder screw.

9) Hold the center of interpreter block.

10) Pull it rightward (a little bit).

11) Push it inward (to the left) as on an axis.

12) By pushing leftward upper side of the block, pull rightward down side of the block (until a 90^{o} turn has occured).

13) After the block has been released remove it from its place and put it on upper cover of the MFCM.

14) Push on lock handle of the right roller's holder axis.

15) Pull out the roller.

16) Release the carbon ribbon from guide roller.

17) Release the carbon ribbon from left guide roller.

18) Release the carbon ribbon along printing area.

19) Throw away one of the rollers into the basket.

20) Insert a pen or a pencil in the central hole of the second roller (as an axis).

21) Hold the pencil as an axis and pull with the left hand the ribbon through the basket until it empties.

22) Release the carbon ribbon from the roller.

23) Insert a new (full) roller at the left (as described on the MFCM).

24) Put edge of ribbon on right rollers (as described on the MFCM).

25) Put edge of ribbon along printing area (as described on the MFCM).

26) Put edge of ribbon on left rollers (as described on the MFCM).

27) Return ribbon's holder of the empty roller (as described on the MFCM).

28) Insert a new left roller (as described on the MFCM).

29) Hold interpreting block.

30) Insert interpreting block according to last stage when removed.

31) Steer axis on their basis.

32) Raise the block 90^{o} (very carefully).

33) Release right screw.

34) Turn lever maximum to the center of the block.

35) Tighten screw.

36) Release left block.

37) Insert the axis to its place.

38) Turn lever maximum to center of the block.

39) Tighten screw.

40) Check to see if the block is steady.

41) Check to see if the ribbon was set as described on the MFCM.

42) Close upper cover.

REMOVAL OF JAMMED CARDS IN THE MFCM

1) Stand behind the MFCM.

2) Remove all cards and papers from MFCM's upper covers.

3) Raise up right upper cover.

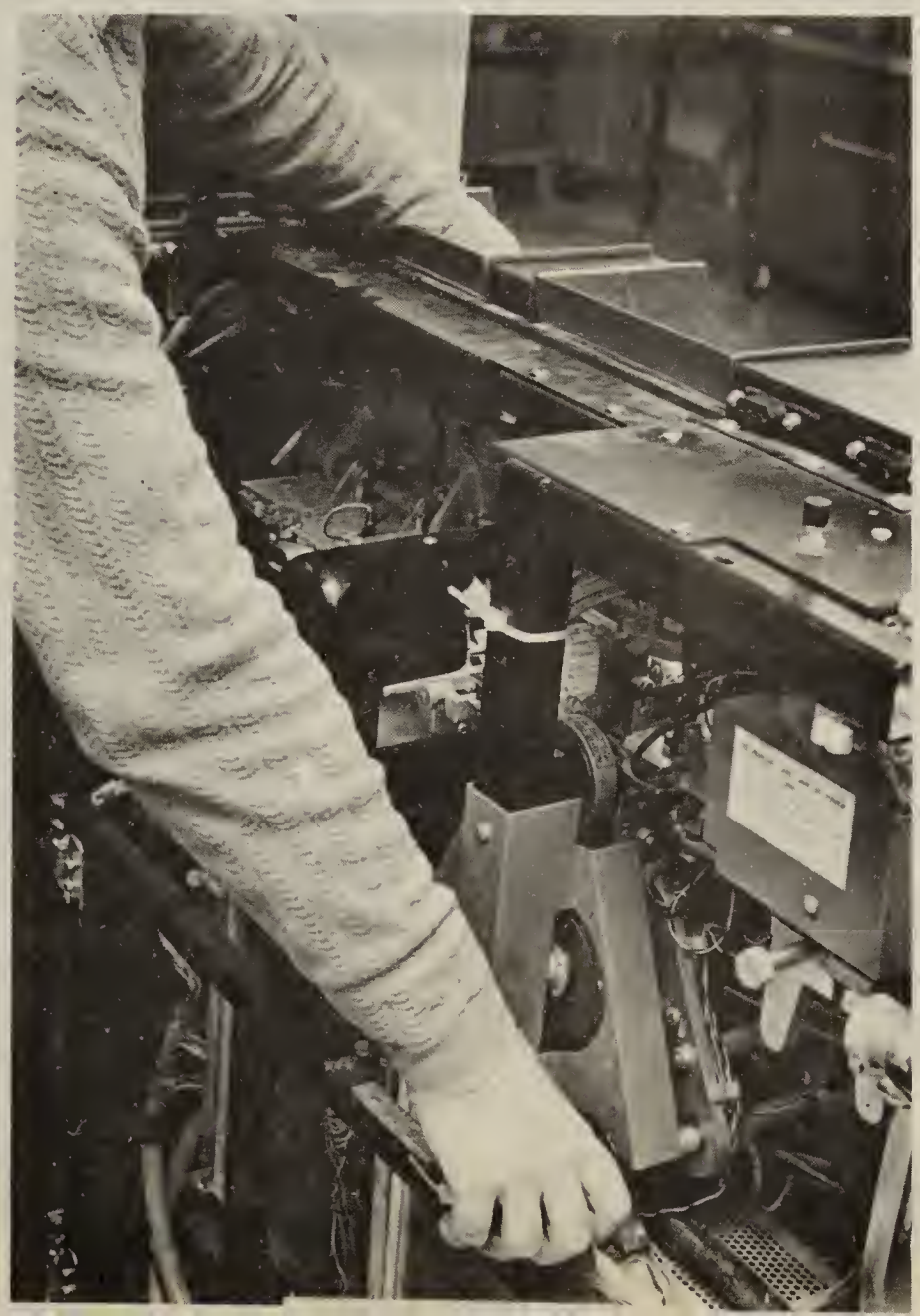
4) Raise up left upper cover.

5) Open upper stacker's cover

6) Open right cover to the right.

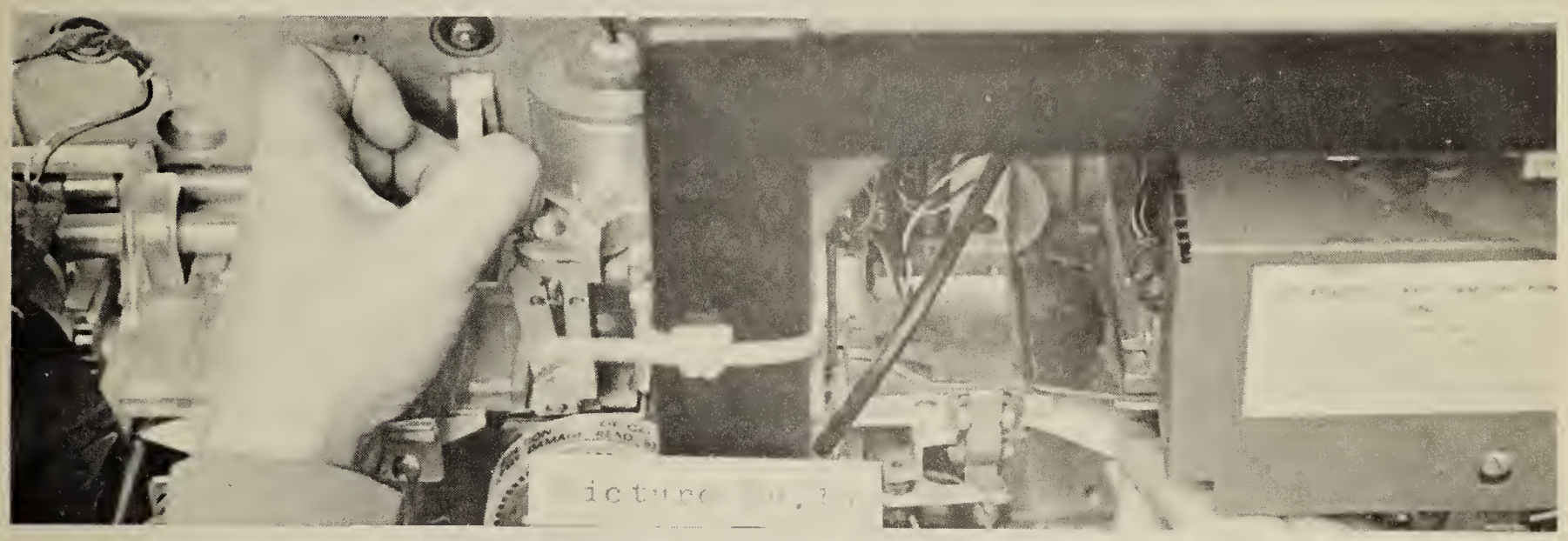
7) Open left cover to the left.

8) Check if there is a jammed card in the stackers. If there is, try to remove it by pulling it very gently. If it is stuck firm turn handle of rollers toward stackers.

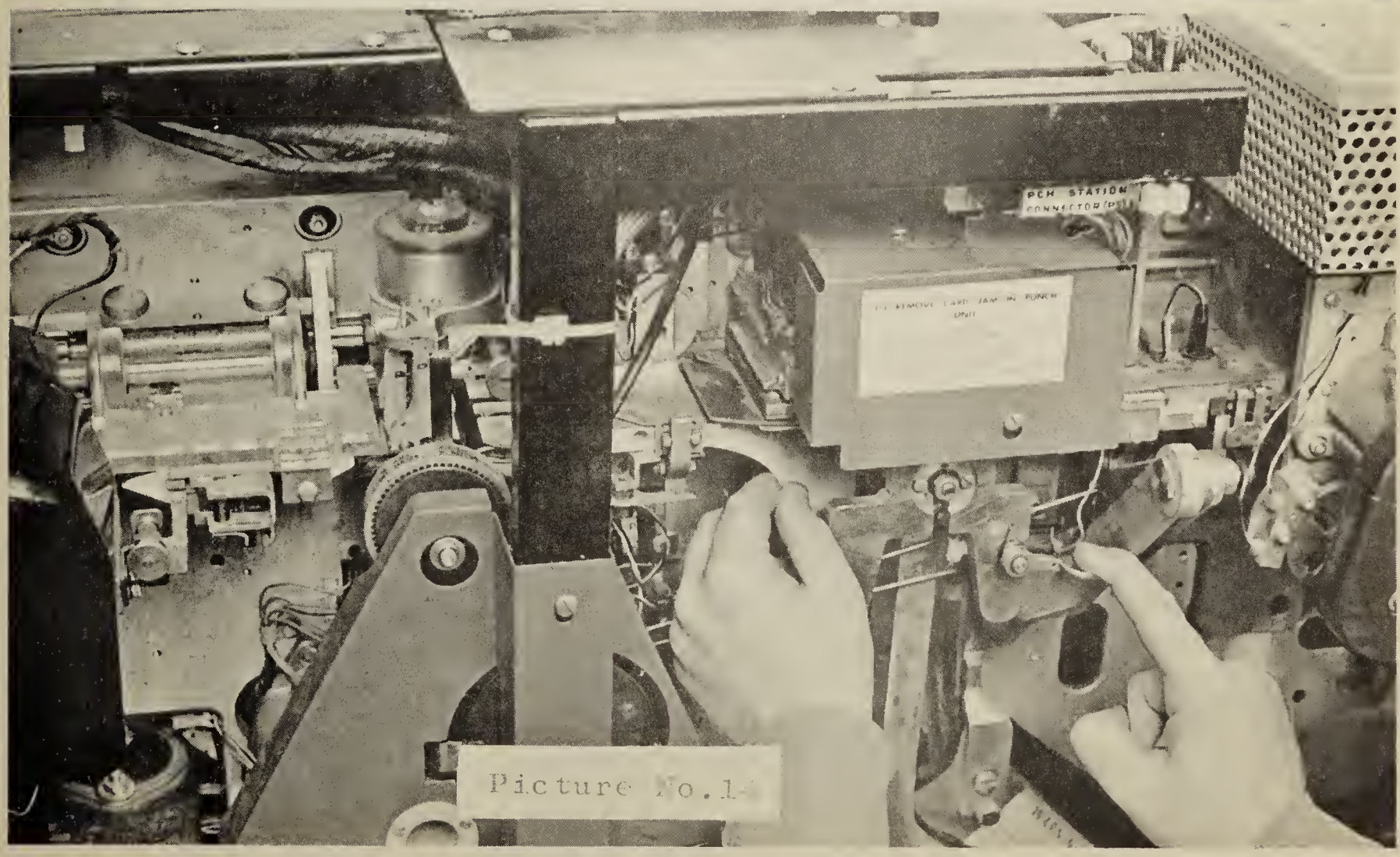


9) If the stackers are empty of jammed cards, check the stations which are covered with plastic doors. Raise up plastic doors, hold it with one hand very carefully.

10) If the jammed card is in the pre-punch punch station, read the notes on the punch unit, then open punch basis screw, pull towards you punch basis, remove jammed cards by pressing on clutch levers. (See two pictures.) Return punch basis to its place and close punch basis screw.



Picture No.1



Picture No.1

11) If the jammed card is in the interpreting station, remove interpreting block (see notes on how to change carbon ribbon).

12) Remove jammed card. Return interpreting block. (See notes on how to change carbon ribbon.)

NOTE: Running the computer while jam removing and afterwards should be done according to orders written in the operating manual of I.B.M.